SECURED ACCESS DEVICE WITH CHIP CARD APPLICATIONS

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to a secured access device with chip card applications.

More specifically, the invention relates to a device for secured access to chip card applications that uses especially instructions which, at each instant, provide information on rights, especially in terms of access to the memory of the chip card, the software component or the hardware operation that has been performed in the chip card.

2. Description of the Prior Art

The most common type of chip card has a microprocessor that manages a program memory. The program memory is usually dedicated to a single application or a set of applications loaded at the same time into the chip card. When several applications are loaded into a chip card, they have a close relationship with one another and are all designed for one and the same type of service. Thus, for example, a chip card cannot simultaneously play the role of a bank card and that of a customer loyalty card for a business of any kind.

In order to end this situation where each chip card has to be limited to one type of application, new software architectures are being considered. These new software architectures are making use of the development of standardized programming languages (for example the language "JAVA") which resolve the problems of portability.

Figure 1 is a simplified view of a software architecture of the chip card projects that are now being developed. The architecture shown in Figure 1 comprises, in particular, a first part 110 that corresponds to what is called the software architecture of a chip card 100 and a second part 120 that corresponds to what is called the applications part of the software architecture of the chip card 100. The system part 110 of the chip card is essentially formed by a library of programs 112 of the chip card operating system, an interface 114 to manage the interactions with, for example, the microprocessor of the chip card or else the different memories of the chip card and a space for the management of hardware interruptions 116.

The applications part 120 of the software architecture consists of different applications:

- a first, second and third main application, respectively 122, 124 and 126;

- a first, second and third additional application, respectively 121, 123 and 125.

The main applications 122, 124 and 126 are written in a programming language that can be directly understood by the processor of the chip card.

The additional applications 121, 123 and 125 are typically applications encoded in a standardized language. These applications may be added at any point in time to the system part 110 in an applications part 120 of the software architecture described. In Figure 1, the additional applications 121, 123 and 125 depend directly on the first main application 122. The first main application 122 herein serves as an interpreter between the additional applications and the operating system by converting the codes of the additional applications into a machine language that can be understood by the programs of the operating system 112.

The device with secured access to applications of a chip card according to the invention comes into play in an architecture of this type.

The software architecture that has just been described is more complex than the one currently existing in chip cards in circulation. Indeed, the architecture described assumes that it is possible to add applications in a standardized programming language, possibly after the chip card is put into circulation. It is therefore more complicated to achieve a satisfactory level of security than was the case when a single application or a group of applications dedicated to a single chip card function was loaded once and for all into the chip card which was then permanently limited in terms of available applications. The risk that a new application might disturb the working of previous applications was therefore not as great.

The coexistence of applications of different kinds in one and the same chip card may raise a certain number of problems. For example, a software architecture simultaneously containing an application dedicated to the assessment of a customer's loyalty to a gasoline company and a standard banking application must ensure that a secret key used in the banking application cannot be read during the use of the application associated with the gasoline company.

SUMMARY OF THE INVENTION

It is an object of the present invention to overcome the problems that have just been described.

To this end, the invention proposes a device enabling the management of different software applications that are installed possibly at different times, or different hardware events, of a chip card while providing for high security. Thus, the device according to the invention offers the possibility of detection when the user of an application tries to exceed his rights for example by attempting to access data not intended for the application in question.

To achieve these goals, the invention proposes to set up specific instructions internal to the microprocessor of the chip card. These specific instructions are call instructions (DCALL) and return instructions (DRETURN). These call and return instructions are associated according to the invention with specific registers by which it can be ascertained that the operations performed by the application during execution in the chip card are authorized or not authorized.

The invention therefore pertains to a device for access to applications of a chip card comprising a microprocessor associated with an operating system working with a set of instructions, a program memory and a battery of applications in a memory of the chip card, wherein the device comprises:

- a register of the microprocessor to store a code, on several check bits, proper to an entity brought into play,

- a call instruction and an instruction for the return of the set of instructions to instantaneously and automatically update the register during the action by a new entity,

- a checking device for the checking, as a function of the check bits, of the authorized character of the access to the zones of the memory of the chip card by the new entity that is called or comes into action in the chip card,

- a first link to transmit the check bits from the microprocessor to the checking device.

According to a particular embodiment of the device of the invention, each new entity taking action is activated at a predefined address of a ROM (read-only memory) type memory of the chip card.

According to different embodiments of the invention, the entity working in the chip card may be an application of the battery of applications or a hardware event, or again the operating system associated with the microprocessor of the chip card.

BRIEF DESCRIPTION OF THE DRAWINGS

The various aspects and advantages of the invention shall appear more clearly hereinafter in the following description made with reference to the appended figures which are given purely by way of an indication and in no way restrict the scope of the invention and which are now introduced:

- Figure 1, already described, is a simplified view of a software architecture of the chip card projects currently being developed,

- Figure 2 is a depiction of the principle of operation according to the invention during the execution of an application within the chip card.

In Figure 2, a microprocessor 200 of a chip card 100 manages the set of operations of a battery of applications 210 of the chip card 100.

MORE DETAILED DESCRIPTION

A two-way bus 250 exchanges information between the microprocessor 200 and any application of the battery of applications 210. The information exchanged may be data elements, addresses or control instructions. A controller of access to the memory 220 exchanges information with the microprocessor 200, especially by means of a link 230 which conveys a signal, called a control signal between the microprocessor 200 and the controller providing access to the memory 220.

For example, when an entity such as the application 211, by means of a two-way bus 250, requires the intervention of another entity such as an application 212, it sends a call instruction DCALL followed by a designation of the entity called and a parameter enabling the nature of the call to be determined. According to the invention, a register R is updated during such calls. A certain number of bits of the register R then assume a value associated with the called entity. The register R is therefore a hardware means of the microprocessor 200 used to store a code proper to the entity of the software architecture that is being performed, and to control its field of execution.

Furthermore, the device according to the invention may also take account of instructions known as hardware instructions, for example instructions of the resetting type. Instructions known as hardware instructions are events that may occur in real time on a chip card and generate interruptions in the microprocessors of the chip cards. This type of event is managed by the device according to the invention in the same way as the software instructions: the bits of the register R take a very

precise value, appropriate to each real-time event that acts on the chip cards, thus limiting and controlling the rights pertaining to these events.

The information given by the register R is thus capable of checking a piece of information, for example at the microprocessor or any other entity external to the software architecture, on the identification of the zone of the software architecture concerned by the application being executed.

The information given by the register R enables the checking of the zone of the memory of the chip card in which the application is entitled to come into action, namely the memory space that it is permitted to access. Thus, any user attempting to make fraudulent use of the operating system in order to recover data pertaining to a particular application is refused access to this data. Indeed, the bits of the state register in this case are different from the bits that might correspond to a call DCALL of the particular application in question. The addresses which it is sought to access and the bits of the register R, sent by the microprocessor by means of the link 230, are compared with each other in the controller of access to the memory 220. Should it be the case that the addresses of the memory that it is sought to access are not addresses belonging to the authorized field of the last application having performed a DCALL type call, then a piece of information on illegal access prohibits access to these memories.

The device according to the invention thus provides great security in the sense that data elements destined for one application cannot be used by another application.

A second register CS makes it possible to retain in memory a code proper to the applications that were active at the last call instruction DCALL sent by the current application, namely those that are to be performed following the current application.

When the current application has finished being executed, a return instruction DRET is executed by the microprocessor and the data elements contained in the second register CS enable a return to the application that was being performed previously and had been activated by a call DCALL. The register R is also updated.

The second register CS cannot be directly accessed by the applications of the chip card. This is in order to ensure the integrity of the device when it is put into operation during the execution of a return instruction DRET.

When the execution of the current application is finished, the bits of the register R assume a value specific to the application that was being performed previously, restoring its rights and limits in terms of memory access.

The memory zone access device according to the invention gives a high level of security in terms of access to the different zones of the memory, for a software architecture such as the one shown in Figure 1.